
SSSSSSSSSSS YYY YYY SSSSSSSSSSS
SSSSSSSSSSS YYY YYY SSSSSSSSSSS
SSSSSSSSSSS YYY YYY SSSSSSSSSSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSSSSSSSS YYY SSSSSSSSS
SSSSSSSSS YYY SSSSSSSSS
SSSSSSSSS YYY SSSSSSSSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSSSSSSSSSS YYY SSSSSSSSSSS
SSSSSSSSSSS YYY SSSSSSSSSSS
SSSSSSSSSSS YYY SSSSSSSSSSS

```
MM      MM  DDDDDDDD      AAAAAA    TTTTTTTTTT
MM      MM  DDDDDDDD      AAAAAA    TTTTTTTTTT
MMMM  MMMM  DD      DD  AA      AA      TT
MMMM  MMMM  DD      DD  AA      AA      TT
MM  MM  MM  DD      DD  AA      AA      TT
MM  MM  MM  DD      DD  AA      AA      TT
MM      MM  DD      DD  AA      AA      TT
MM      MM  DD      DD  AA      AA      TT
MM      MM  DD      DD  AAAAAAAAAA      TT
MM      MM  DD      DD  AAAAAAAAAA      TT
MM      MM  DD      DD  AA      AA      TT          ....
MM      MM  DD      DD  AA      AA      TT          ....
MM      MM  DDDDDDDD    AA      AA      TT          ....
MM      MM  DDDDDDDD    AA      AA      TT          ....

LL              IIIIII      SSSSSSSS
LL              IIIIII      SSSSSSSS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II        SSSSSS
LL                II        SSSSSS
LL                II              SS
LL                II              SS
LL                II              SS
LL                II              SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

Table of contents

```
0000     1          .IF     NDF,PRMSW       ;
0000     2          .TITLE  MDAT            ;MEMORY MANAGEMENT DATA BASE
0000     3          .IFF                    ;
0000     4          .TITLE  SPTSKEL - SKELETON SYSTEM PAGE TABLE
0000     5          .ENDC                   ;
0000     6          .IDENT  'V04-000'
0000     7  ;
0000     8  ;*****************************************************************************
0000     9  ;*                                                                           *
0000    10  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000    11  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
0000    12  ;*  ALL RIGHTS RESERVED.                                                     *
0000    13  ;*                                                                           *
0000    14  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
0000    15  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
0000    16  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
0000    17  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
0000    18  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
0000    19  ;*  TRANSFERRED.                                                             *
0000    20  ;*                                                                           *
0000    21  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
0000    22  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0000    23  ;*  CORPORATION.                                                             *
0000    24  ;*                                                                           *
0000    25  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
0000    26  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    27  ;*                                                                           *
0000    28  ;*                                                                           *
0000    29  ;*****************************************************************************
0000    30
0000    31  ;++
0000    32  ; FACILITY:     EXECUTIVE,  MEMORY MANAGEMENT DATA BASE
0000    33  ;
0000    34  ; ABSTRACT:     MDAT ALLOCATES AND INITIALIZES THE STORAGE FOR THE
0000    35  ;       MEMORY MANAGEMENT DATA BASES. IT IS ASSEMBLED IN TWO FORMS
0000    36  ;       ONE TO PRODUCE A SKELETON SPT AND THE OTHER TO PRODUCE THE SYSTEM
0000    37  ;       MEMORY MANAGEMENT DATA STRUCTURES.
0000    38  ;
0000    39  ; ENVIRONMENT:
0000    40  ;
0000    41  ;--
0000    42  ;
0000    43          .SBTTL  HISTORY                 ; DETAILED
0000    44  ;
0000    45  ; AUTHOR: RICHARD I. HUSTVEDT   , CREATION DATE: 18-MAY-1978
0000    46  ;
0000    47  ; MODIFIED BY:
0000    48  ;
0000    49  ;       V03-007 WHM0001         Bill Matthews           02-May-1984
0000    50  ;               Make PAT$A_NONPGD_CODE_END global for use by SYSBOOT to
0000    51  ;               initial MMG$GL_PGDCOD.
0000    52  ;
0000    53  ;       V03-006 LJK0273         Lawrence J. Kenah       10-Apr-1984
0000    54  ;               Only set a single page to UREW to hold file system statistics.
0000    55  ;               Add cells to hold base addresses of various loadable images.
0000    56  ;               Remove cells added for MWAIT measurements.
0000    57  ;
```

```
0000     58 ;       V03-005 RLRSCORP        Robert L. Rappaport     17-Feb-1984
0000     59 ;               Added EXE$GL_CPUNODSP, a pointer to the virtual address
0000     60 ;               that maps node private space on a BI. For a BI processor,
0000     61 ;               such as KDZ-11, this allows access to processor internal
0000     62 ;               registers.  Also added EXE$GQ_GBLHOOK1 - EXE$GQ_GBLHOOKA,
0000     63 ;               global symbols each of which defines a quadword of data.
0000     64 ;               These can be used as hooks to facilitate adding support for
0000     65 ;               new hardware between major releases.  Also add three more
0000     66 ;               pages of extra patch area for a total of six such pages.
0000     67 ;
0000     68 ;       V03-004 KPL0101         Peter Lieberwirth       1-Feb-1984
0000     69 ;               I was talked into changing CONFREG1 to CONFREGL, since
0000     70 ;               CONFREGL is a more descriptive name.
0000     71 ;
0000     72 ;       V03-003 KPL0100         Peter Lieberwirth       30-Jan-1984
0000     73 ;               Add cell to point to new CONFREG array, called CONFREG1.
0000     74 ;               Eventually, all references in the system to CONFREG will
0000     75 ;               be changed to refer to the new format CONFREG1.  At that
0000     76 ;               time, the extra CONFREG cell can be deleted.
0000     77 ;
0000     78 ;       V03-002 SSA0005         Stan Amway              10-Jan-1984
0000     79 ;               Reserved 148 bytes in non-paged data patch area for
0000     80 ;               special MWAIT counters being maintained by code in
0000     81 ;               module MUTEX. This change will be backed out before
0000     82 ;               V4 release.
0000     83 ;
0000     84 ;       V03-001 LJK0159         Lawrence J. Kenah       9-Apr-1982
0000     85 ;               Include holes caused by page alignment into patch areas.
0000     86 ;               Change names of PSECTs and global labels to include
0000     87 ;               string "PATCH".
0000     88 ;
0000     89 ;       V02-013 LJK0095         Lawrence J. Kenah       3-Dec-1981
0000     90 ;               Move definition of label that marks boundary between
0000     91 ;               nonpaged and pageable executive to SYSPARAM so that
0000     92 ;               cell containing the boundary is accessible to SYSBOOT.
0000     93 ;
0000     94 ;       V02-012 LJK0078         Lawrence J. Kenah       6-Nov-1981
0000     95 ;               Increase size of read-only (pageable and nonpaged) patch
0000     96 ;               areas to two pages each. Add two more pages that can be
0000     97 ;               used for either pageable or nonpaged patch area.
0000     98 ;
0000     99 ;       V02-011 LJK0074         Lawrence J. Kenah       6-Oct-1981
0000    100 ;               Point MMG$GL_RMSBASE to procedure that always returns success.
0000    101 ;               This prevents anamolous system failures when RMS is called
0000    102 ;               inadvertently before the RMS image is mapped.
0000    103 ;
0000    104 ;       V02-010 WMC0002         Wayne Cardoza           20-Aug-1981
0000    105 ;               Add MMG$GL_GBLPAGFIL to limit page file utilization for global
0000    106 ;               sections with page file backing store.
0000    107 ;
0000    108 ;       V02-009 WMC0001         Wayne Cardoza           12-Aug-1981
0000    109 ;               Add MMG$GL_GBLSECFND to assist in finding section tables for
0000    110 ;               global sections with page file backing store.
0000    111 ;
0000    112 ;       V02-008 HRJ0023         Herb Jacobs             06-Jul-1981
0000    113 ;               Indicate system process doesn't need swap space.
0000    114 ;
```

```
0000   115 ;        V02-007 LJK0030         Lawrence J. Kenah       28-May-1981
0000   116 ;                Add global labels for three arrays used by INIT for opcode
0000   117 ;                fixup that occurs at bootstrap time.
0000   118 ;
0000   119 ;        V02-006 HRJ0021         Herb Jacobs             10-May-1981
0000   120 ;                Fix historic reference to WSNEXT-1 to WSNEXT.
0000   121 ;
0000   122 ;        V02-005 TCM0001         Trudy C. Matthews       8-May-1981
0000   123 ;                Delete the definition of MMG$AL_SBICONF array.  Instead add
0000   124 ;                EXE$GL_CONFREG and MMG$GL_SBICONF, which hold the addresses
0000   125 ;                of the arrays (which are allocated in pool).
0000   126 ;                Add definition of EXE$GL_NUMNEXUS field, to hold number of
0000   127 ;                nexuses present on the system.
0000   128 ;
```

```
                     0000   130          .SBTTL  DECLARATIONS
                     0000   131
                     0000   132 ;
                     0000   133 ; INCLUDE FILES:
                     0000   134 ;
                     0000   135          $DYNDEF                 ;DYNAMIC DATA STRUCTURE TYPE DEFINITIONS
                     0000   136          $PHDDEF                 ;DEFINE PROCESS HEADER
                     0000   137          $PTEDEF                 ;PAGE TABLE ENTRY DEFINTIONS
                     0000   138          $SECDEF                 ;PSTE/GSTE DEFINITIONS
                     0000   139          $SGNDEF                 ;DEFINE SYSGEN VALUES
                     0000   140          $WSLDEF                 ;WORKING SET LIST DEFINITIONS
                     0000   141 ;
                     0000   142 ; EXTERNAL SYMBOLS:
                     0000   143 ;
                     0000   144
                     0000   145 ;
                     0000   146 ; MACROS:
                     0000   147 ;
                     0000   148          .MACRO  SYSPTE  NUM,ACCESS,PFN=0
                     0000   149          .IF     DF,PRMSW
                     0000   150          .PSECT  $$$065
                     0000   151          .ENDC
                     0000   152          .REPT   NUM
                     0000   153          .IF     DF,PRMSW
                     0000   154          .LONG   PTE$M_VALID!PTE$C_'ACCESS
                     0000   155          .ENDC
                     0000   156          PFN...=PFN...+1
                     0000   157          SPTLEN=SPTLEN+1
                     0000   158          .ENDR
                     0000   159          .ENDM   SYSPTE
                     0000   160
                     0000   161          .MACRO  PHD     SYM
                     0000   162          .=SAV...+PHD$'SYM
                     0000   163          .ENDM   PHD
                     0000   164
                     0000   165          .MACRO  PCB     SYM
                     0000   166          .=SAV...+PCB$'SYM
                     0000   167          .ENDM   PCB
                     0000   168
                     0000   169          .LIST   MEB
                     0000   170 ;
                     0000   171 ; EQUATED SYMBOLS:
                     0000   172 ;
000001F8             0000   173          NPGDPATCH = 504                 ; ONE PAGE OF NONPAGED CODE PATCH AREA
000001F8             0000   174          NPGDRWPATCH = 504                ; ONE PAGE OF NONPAGED DATA PATCH AREA
000003F8             0000   175          PGDPATCH = 504 + 512             ; TWO PAGES OF PAGED CODE PATCH AREA
00000C00             0000   176          PATCH_AREA = 6*512               ; SIX PAGES OF EXTRA PATCH AREA
                     0000   177 ;
                     0000   178 ; OWN STORAGE:
                     0000   179 ;
                     0000   180
```

```
                     0000   182          .SBTTL  MEMORY MANAGEMENT DATA BASE
                     0000   183
                     0000   184          .IF     NDF,PRMSW               ;
                     0000   185 ;
                     0000   186 ; PROCESS HEADER VECTOR
                     0000   187 ;
                00000000    0000   188          .PSECT  $$$222,LONG
                     0000   189 PHV$GL_PIXBAS::                          ;BASE OF PROCESS INDEX VECTOR
            00000000 0000   190          .LONG   0                      ;
                     0004   191 PHV$GL_REFCBAS::                         ;BASE OF PROCESS HDR REFERENCE COUNT VECTOR
            00000000 0004   192          .LONG   0                      ;
                     0008   193
                     0008   194 ;
                     0008   195 ; Define Global Hooks
                     0008   196 ;
                     0008   197
                     0008   198 EXE$GQ_GBLHOOK1::
      00000000 00000000 0008   199          .QUAD   0
                     0010   200 EXE$GQ_GBLHOOK2::
      00000000 00000000 0010   201          .QUAD   0
                     0018   202 EXE$GQ_GBLHOOK3::
      00000000 00000000 0018   203          .QUAD   0
                     0020   204 EXE$GQ_GBLHOOK4::
      00000000 00000000 0020   205          .QUAD   0
                     0028   206 EXE$GQ_GBLHOOK5::
      00000000 00000000 0028   207          .QUAD   0
                     0030   208 EXE$GQ_GBLHOOK6::
      00000000 00000000 0030   209          .QUAD   0
                     0038   210 EXE$GQ_GBLHOOK7::
      00000000 00000000 0038   211          .QUAD   0
                     0040   212 EXE$GQ_GBLHOOK8::
      00000000 00000000 0040   213          .QUAD   0
                     0048   214 EXE$GQ_GBLHOOK9::
      00000000 00000000 0048   215          .QUAD   0
                     0050   216 EXE$GQ_GBLHOOKA::
      00000000 00000000 0050   217          .QUAD   0
                     0058   218 ;
                     0058   219 ; Define data to identify the nexus on a system.
                     0058   220 ;
                     0058   221 EXE$GL_CPUNODSP::                        ; Holds virtual address that maps BI
            00000000 0058   222          .LONG   0                      ;  Node Private Space.  Used only for
                     005C   223                                         ;  Scorpio, and allows access to Port
                     005C   224                                         ;  Controler, Watch Chip, and RX50
                     005C   225                                         ;  registers.
                     005C   226 EXE$GL_CONFREGL::                       ; Holds the address of a longword array
            00000000 005C   227          .LONG   0                      ; of nexus device types.
                     0060   228 EXE$GL_CONFREG::                        ; Holds the address of a byte array
            00000000 0060   229          .LONG   0                      ; of nexus-device types.
                     0064   230 MMG$GL_SBICONF::                        ; Holds the address of a longword
            00000000 0064   231          .LONG   0                      ; array of nexus slot VAs.
                     0068   232 EXE$GL_NUMNEXUS::                       ; Number of nexuses present on system.
            00000000 0068   233          .LONG   0
                     006C   234
                     006C   235 ; The following cell contains the base address of the RMS image
                     006C   236
                     006C   237 MMG$GL_RMSBASE::                        ; Base of RMS image
            00000000' 006C   238          .ADDRESS        EXE$SUCCESS    ; This procedure always succeeds
```

```
                     0070   239
                     0070   240 ; The following cells contain the base addresses of various images
                     0070   241 ; that may be loaded when the system is started.
                     0070   242
                     0070   243 MMG$GL_FPEMUL_BASE::                    ; Base address of folating point
00000000             0070   244         .LONG   0                      ;  instruction emulator
                     0074   245
                     0074   246 MMG$GL_SYSLOA_BASE::                    ; Base address of SYSLOAzzz.EXE
00000000             0074   247         .LONG   0
                     0078   248
                     0078   249 MMG$GL_VAXEMUL_BASE::                   ; Base address of decimal/string
00000000             0078   250         .LONG   0                      ;  instruction emulator
                     007C   251
                     007C   252 MMG$GL_GBLSECFND::                     ; Last global section table entry found
00000000             007C   253         .LONG   0                      ; when deleting page file backing store addr
                     0080   254 MMG$GL_GBLPAGFIL::
FFFFFFFF             0080   255         .LONG   -1                     ; page file allowed (remaining) for global s
                     0084   256         .ENDC                          ;
```

```
0084    258          .SBTTL  SYSTEM HEADER AND PAGE TABLE
0084    259 ;-----------------------------------------------------------------------
0084    260 ;
0084    261 ;       SYSTEM HEADER / SYSTEM WORKING SET LIST / SYSTEM PAGE TABLE
0084    262 ;
0084    263 ;-----------------------------------------------------------------------
0084    264          .IF     DF,PRMSW                ;
0084    265          .PSECT  $$$063,PAGE             ; PAGE ALIGNED
0084    266
0084    267 BOO$A_SYSPHD::                           ; SYSTEM PROCESS HEADER
0084    268          SAV...=.                        ; REFERENCE POINT FOR FILLING PHD
0084    269          .BLKB   PHD$C_LENGTH            ; RESERVE SPACE FOR IT
0084    270 SYSPHDEND=.                              ; MARK END OF PHD
0084    271
0084    272 WSL...=<.-SAV...>@-2                     ; LONGWORD INDEX TO FIRST WS ENTRY
0084    273          PHD     W_WSLOCK                ; POINTER TO START OF LOCKED PAGES
0084    274          .WORD   WSL...                  ;
0084    275
0084    276          PHD     W_WSDYN                 ; POINTER TO START OF DYNAMIC WS
0084    277          .WORD   WSL...                  ;
0084    278
0084    279          PHD     W_WSLIST                ; START OF WORKING SET LIST
0084    280          .WORD   WSL...                  ;
0084    281
0084    282          PHD     W_WSNEXT                ; NEXT WORKING SET ENTRY
0084    283          .WORD   WSL...                  ;
0084    284
0084    285          PHD     L_FREP1VA               ; SMALLEST VA IN P1 SPACE (EMPTY)
0084    286          .LONG   -1
0084    287
0084    288          PHD     W_EXTDYNWS              ; EXTRA DYNAMIC WORKING SET LIST
0084    289          .WORD   4096                    ; LARGE NUMBER TO DEFEAT TEST FOR
0084    290
0084    291          PHD     W_SWAPSIZE              ; SWAP SPACE SIZE TO SWAP PROCESS
0084    292          .WORD   -1                      ; DISABLE FOR SYSTEM PROCESS
0084    293
0084    294          PHD     L_PTWSLELCK             ; POINTER TO LOCKED PAGE TABLE ARRAY
0084    295          .LONG   ^X40000000              ; FORCE ACCESS VIOLATION FOR SYSTEM SPACE
0084    296
0084    297          PHD     L_PTWSLEVAL             ; POINTER TO VALID PAGE TABLE ARRAY
0084    298          .LONG   ^X40000000              ; FORCE ACCESS VIOLATION FOR SYSTEM SPACE
0084    299
0084    300 .=SYSPHDEND                              ; RESTORE LOCATION COUNTER
0084    301 SYSPHDLEN=.-SAV...                       ; LENGTH OF SYSTEM HEADER
0084    302          .ENDC                           ;
```

```
                     0084    304          .SBTTL  SYSTEM PAGE TABLE
                     0084    305 ;
                     0084    306 ; BUILD THE SYSTEM PAGE TABLE
                     0084    307 ;
                     0084    308          .IF     DF,PRMSW                        ;
                     0084    309          .PSECT  $$$065,PAGE
                     0084    310          .ENDC                                   ;
           00000000  0084    311          PFN...=0
           00000000  0084    312          SPTLEN=0                                ; INITIALIZE LENGTH COUNTER
                     0084    313          .IF     DF,PRMSW                        ;
                     0084    314 MMG$AL_SYSPAGTB::                               ;SYSTEM VIRTUAL ADDRESS OF SPT
                     0084    315          .ENDC                                   ;
                     0084    316 ;
                     0084    317 ; SYSTEM SERVICE VECTORS - PSECT $$$000 HAS SGN$C_SYSVECPGS PAGES ALLOCATED ELSEWHER
                     0084    318 ;
                     0084    319          SYSPTE  SGN$C_SYSVECPGS,UR              ;SYSTEM SERVICE VECTORS ($$$000)
                     0084    320          SYSPTE  1,UREW                          ;FCP PERFORMANCE DATA PAGE
                     0084    321
                     0084    322          .IF     DF,PRMSW
                     0084    323 MMG$C_SPTSKEL==SPTLEN                           ; LENGTH OF SKELETON SPT IN  LONGWORDS
                     0084    324          .ENDC
                     0084    325
                     0084    326 ;
                     0084    327          .IF     NDF,PRMSW
                     0084    328          .ENDC
                     0084    329          .IF     NDF,PRMSW                       ;
                00000000    330          .PSECT  $$$000ENDVEC,PAGE,EXE
                     0000    331 MMG$A_ENDVEC::                                  ; MARKER FOR END OF VECTOR PAGES
                00000000    332          .PSECT  $$$900,PAGE                     ; MARKER FOR BASE OF SYSPARAM
                     0000    333 MMG$A_SYSPARAM::                                ;
                00000000    334          .PSECT  $$$890_PATCH_NONPGD_DATA,LONG,EXE,WRT
                     0000    335 PAT$A_NONPGD_DATA::                             ; NONPAGED DATA PATCH AREA
           FFFFFFF8'  0000    336          .LONG   MMG$A_SYSPARAM-<.+8>            ; SIZE OF AREA (INCLUDE EXCESS)
           00000008'  0004    337          .ADDRESS        .+4                     ; POINTER TO FIRST AVAILABLE BYTE
           00000200   0008    338          .BLKB   NPGDRWPATCH                     ;
                      0200    339
                00000000    340          .PSECT  $$$999,PAGE,EXE                 ;END OF WRITABLE REGION
                     0000    341 MMG$FRSTRONLY::                                 ;SYSTEM VIRTUAL ADDRESS
                     0000    342                                                 ;OF FIRST READ ONLY PAGE
                     0000    343
```

```
                        .SUBTITLE       READ-ONLY PATCH AREAS

                ;+
                ; There is a single page of read-only patch space located at the boundary
                ; between the nonpaged and pageable exec routines. This page is used for
                ; patches to the nonpaged routines in SYS.EXE. There are two pages located
                ; in the middle of the pageable exec routines that are used for a pageable
                ; patch area.
                ;
                ; In addition, there are three more pages located at the boundary
                ; between the nonpaged and pageable exec routines. These pages are all
                ; initially pageable. If either read-only patch area needs room to
                ; expand, one of these pages can be used.
                ;
                ;    o If a pageable page is required, it should be taken from the
                ;      high address end (the third page). A patch descriptor must
                ;      be added for each page in this area used for pageable patch
                ;      area.
                ;
                ;    o If more nonpaged patch space is needed, that can be obtained
                ;      by extending the current nonpaged patch area. This expansion
                ;      consists of two steps. The first longword in the patch
                ;      descriptor (global label PAT$A_NONPGD_CODE) must be increased by
                ;      512 to reflect the size increase in the patch area. The
                ;      contents of the cell MMG$GL_PGDCOD, the boundary between the
                ;      nonpaged and pageable exec, must be increased by 512 to reflect
                ;      the fact that the nonpaged exec has grown by a page. To simplify
                ;      location of these two cells, they have additional labels that
                ;      clearly relate them to expanding nonpaged read-only patch area.
                ;      MMG$GL_PGDCOD is now loaded from BOO$GL_PGDCOD in SYSBOOT and
                ;      therefore BOO$GL_PGDCOD must be patched with the increased size.
                ;      MMG$GL_PGDCOD will get the increased size on reboot.
                ;
                ;              PAT$A_NONPGD_CODE       PAT$GL_EXP_NPG1
                ;              MMG$GL_PGDCOD           PAT$GL_EXP_NPG2
                ;-

                        .PSECT  X___PATCH_NONPGD_CODE,EXE       ; NONPAGED CODE PATCH AREA
                PAT$A_NONPGD_CODE::                             ; NONPAGED PURE
                PAT$GL_EXP_NPG1::                               ; (SYNONYM)
FFFFFFF8'       .LONG   PAT$A_NONPGD_CODE_END-<.+8>             ; SIZE OF NONPAGED PATCH AREA
00000008'       .ADDRESS        .+4                             ; POINTER TO START
00000200        .BLKB   NPGDPATCH                               ; ALLOCATE PAGE AREA

                ;  The rest of this patch area starts out as pageable exec. It may be
                ;  made part of the nonpaged exec if more than one page of nonpaged
                ;  patch space is needed.

                        .PSECT  Y$$$PATCH_EXTEND_CODE,PAGE
                PAT$A_NONPGD_CODE_END::                         ; END OF NONPAGED PATCH AREA
00000C00        .BLKB   PATCH_AREA

                        .PSECT  YF$$$PATCH_PAGED_CODE,LONG      ; PATCH ARE FOR PAGED CODE

                ;  The pageable read-only patch area is placed approximately in the middle
                ;  of the pageable exec to allow control to be passed into and out of the
                ;  patch area with BRW instructions rather than JMP instructions.
```

```
                0000   402
                0000   403 PAT$A_PAGED_CODE::                                ;
000003F8        0000   404         .LONG   PGDPATCH                          ; SIZE OF AREA
00000008'       0004   405         .ADDRESS        .+4                       ; START OF FREE AREA
00000400        0008   406         .BLKB   PGDPATCH                          ;
                0400   407
                0400   408 ;
                0400   409 ;       MARK END OF PAGED CODE
                0400   410 ;
            00000000   411         .PSECT  YZ99$PAGEDEND,PAGE
                0000   412 MMG$AL_PGDCODEN::                                  ;
                0000   413
```

```
                        .SUBTITLE       OTHER GLOBAL LABELS

                ;
                ; DEFINE BEGINNING AND END OF DRIVER REGION
                ;

                        .PSECT  $$$110_BEGDRIVE,LONG
MMG$AL_BEGDRIVE::                               ;
                        .PSECT  $$$120_ENDDRIVE,LONG
MMG$AL_ENDDRIVE::                               ;

                ;
                ; Define global labels for opcode/address table used by fixup code in
                ; INIT when more than 32 Mbytes of memory is present on the system.
                ; Each six byte entry in this table consists of an address whose contents
                ; are to be altered, a byte containing the current contants of that location
                ; to be used as a sanity check, and a byte containing the new opcode. The
                ; table is terminated with an address of zero.
                ;

                        .PSECT  Z$INIT$PFN_FIXUP_TABLE
MMG$AL_FIXUPTBL::                               ; Listhead for opcode/address table

                        .ENDC                   ;
                        .END
```

```
EXE$GL_CONFREG                  00000060 RG    02
EXE$GL_CONFREGL                 0000005C RG    02
EXE$GL_CPUNODSP                 00000058 RG    02
EXE$GL_NUMNEXUS                 00000068 RG    02
EXE$GQ_GBLHOOK1                 00000008 RG    02
EXE$GQ_GBLHOOK2                 00000010 RG    02
EXE$GQ_GBLHOOK3                 00000018 RG    02
EXE$GQ_GBLHOOK4                 00000020 RG    02
EXE$GQ_GBLHOOK5                 00000028 RG    02
EXE$GQ_GBLHOOK6                 00000030 RG    02
EXE$GQ_GBLHOOK7                 00000038 RG    02
EXE$GQ_GBLHOOK8                 00000040 RG    02
EXE$GQ_GBLHOOK9                 00000048 RG    02
EXE$GQ_GBLHOOKA                 00000050 RG    02
EXE$SUCCESS                     ******** X     02
MMG$AL_BEGDRIVE                 00000000 RG    0B
MMG$AL_ENDDRIVE                 00000000 RG    0C
MMG$AL_FIXUPTBL                 00000000 RG    0D
MMG$AL_PGDCODEN                 00000000 RG    0A
MMG$A_ENDVEC                    00000000 RG    03
MMG$A_SYSPARAM                  00000000 RG    04
MMG$FRSTRONLY                   00000000 RG    06
MMG$GL_FPEMUL_BASE              00000070 RG    02
MMG$GL_GBLPAGFIL                00000080 RG    02
MMG$GL_GBLSECFND                0000007C RG    02
MMG$GL_RMSBASE                  0000006C RG    02
MMG$GL_SBICONF                  00000064 RG    02
MMG$GL_SYSLOA_BASE              00000074 RG    02
MMG$GL_VAXEMUL_BASE             00000078 RG    02
NPGDPATCH                     = 000001F8
NPGDRWPATCH                   = 000001F8
PAT$A_NONPGD_CODE               00000000 RG    07
PAT$A_NONPGD_CODE_END           00000000 RG    08
PAT$A_NONPGD_DATA               00000000 RG    05
PAT$A_PAGED_CODE                00000000 RG    09
PAT$GL_EXP_NPG1                 00000000 RG    07
PATCH_AREA                    = 00000C00
PFN...                        = 00000006
PGDPATCH                      = 000003F8
PHV$GL_PIXBAS                   00000000 RG    02
PHV$GL_REFCBAS                  00000004 RG    02
SGN$C_SYSVECPGS               = 00000005
SPTLEN                        = 00000006
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
$$$222	00000084 (132.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG
$$$000ENDVEC	00000000 (0.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC PAGE
$$$900	00000000 (0.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC PAGE
$$$890_PATCH_NONPGD_DATA	00000200 (512.)	05 (5.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG
$$$999	00000000 (0.)	06 (6.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC PAGE
X___PATCH_NONPGD_CODE	00000200 (512.)	07 (7.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
Y$$$PATCH_EXTEND_CODE	00000C00 (3072.)	08 (8.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC PAGE
YF$$$PATCH_PAGED_CODE	00000400 (1024.)	09 (9.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG
YZ99$PAGEDEND	00000000 (0.)	0A (10.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC PAGE
$$$110_BEGDRIVE	00000000 (0.)	0B (11.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG
$$$120_ENDDRIVE	00000000 (0.)	0C (12.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG
Z$INIT$PFN_FIXUP_TABLE	00000000 (0.)	0D (13.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.05	00:00:02.05
Command processing	117	00:00:00.49	00:00:06.21
Pass 1	193	00:00:04.57	00:00:14.72
Symbol table sort	0	00:00:00.57	00:00:01.40
Pass 2	93	00:00:01.26	00:00:05.06
Symbol table output	6	00:00:00.08	00:00:00.36
Psect synopsis output	4	00:00:00.06	00:00:00.06
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	450	00:00:07.08	00:00:29.86

The working set limit was 1350 pages.
26101 bytes (51 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 456 non-local and 0 local symbols.
439 source lines were read in Pass 1, producing 36 object records in Pass 2.
16 pages of virtual memory were used to define 15 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYS.OBJ]LIB.MLB;1	5
_255DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	9

493 GETS were required to define 9 macros.

There were no errors, warnings or information messages.

```
MACRO/LIS=LIS$:MDAT/OBJ=OBJ$:MDAT MSRC$:MDAT/UPDATE=(ENH$:MDAT)+EXECML$/LIB
```